# CONTROL SYSTEM UPGRADE FOR SUPERKEKB INJECTOR LINAC

M. Satoh[#], Y. Seimiya, F. Miyahara, K. Mikawa, T. Suwada, and K. Furukawa, KEK/ SOKENDAI, Tsukuba, Japan
T. Kudou, S. Kusano, Mitsubishi Electric System & Service Co., Ltd, Tsukuba, Japan
H. Saotome, M. Takagi, Kanto Information Service (KIS), Accelerator Group, Tsuchiura, Japan

*Abstract*

The KEKB project has successfully completed its decade operation in the June of 2010. SuperKEKB main ring is currently being constructed for aiming at the peak luminosity of 8 x $10^{35}$ $cm^{-2}s^{-1}$. The electron/positron injector linac upgrade is also going on for increasing the intensity of bunched charge with keeping the small emittance. The key upgrade issues are the construction of positron damping ring, a new positron capture system, and a low emittance photo-cathode rf electron source. The injector linac beam commissioning started in the October of 2013. The whole control system performance determines the beam operation efficiency of injector linac, and it eventually has a strong impact on the experimental results of physics. In this decade, the linac control system has gradually transferred from the in-house system to the Experimental Physics and Industrial Control System based one for increasing the availability of beam operation. In this paper, we present the detail of SuperKEKB injector linac control system.

## INTRODUCTION

The linac beam control system is based on a standard client/server model with three levels. The original linac control system has been developed by using the in-house software libraries based on the remote procedure call (RPC). The client user interfaces have been designed by a command line interface and Tcl/Tk scripting language. Around a decade ago, the middle phase of KEKB operation, the linac control system has been upgraded to a new framework based-on the Experimental Physics and Industrial Control System (EPICS) [1] to improve the affinity between the linac and ring control systems. These improvements make it easy to analyze the correlations between the linac and ring parameters. Such analysis can help finding a source of injection rate deterioration. The new client user interfaces have been implemented by Python scripting language for the rapid software development. For the simultaneous top-up injection of KEB and PF rings, an event based timing system has been implemented to enable the pulse-to-pulse beam modulation [2].

## CONTROL SYSTEM OVERVIEW

### *Server and client computer environment*

The linac control system is based on the server/client model. It is useful framework to integrate the different kinds of local controllers. In the beginning of KEB project, we used the six Compaq Alpha servers with the Tru64 UNIX operating system as the server computers. Two of them were connected to the RAID disk via SCSI bus interface, and they can work as the active/standby redundant NFS servers for the high availability operation. Since the Tru64 UNIX was obsoleted, they have been gradually replaced by the Linux-based machines. After some different types of high availability cluster system based on Linux were evaluated, eventually we made decision to use the Linux base system without cluster functionality. Instead of redundant system, the blade based server system was employed as the high reliability server systems. Currently, eleven Linux based blade servers are utilized as the server computers. Both of server and client side control software are running on the server machines. Currently, Linux distribution of CentOS 5.11 x86_64 are mainly used for the daily operation. CentOS 6.6 and 7 are also under test for the future utilization. The Tektronix X terminals and touch panel displays based on PC9801/DOS machines have been originally utilized as the operator terminals. These terminals were replaced by the Linux PC and Windows PC with the X server application software of ASTEC-X and Reflection X.

### *Network environment*

The network system is one of the most important infrastructures for the reliable accelerator control system. During the KEKB operation, Cisco Catalyst 4506 and 3750 were utilized as the core switches. Each of them were independently connected to 45 edge switches of Catalyst 2950 via optical fiber with 100 Mbps bandwidth. The two core switches worked as the active/standby redundant system. For the network connection between the edge switches and local controllers like programmable logic controllers (PLCs), the optical fiber is used for avoiding the noise generated by the klystron modulators.

Toward SuperKEKB, the core switch system was replaced by 6 of Cisco Catalyst 3750X. Each of them can work as the active/active redundant system based on the virtual switching system technology. We replaced also the edge switch by Catalyst 2960S, and the network connection between them was improved up to 1 Gbps bandwidth.

### *Local controllers*

For the injector linac control system, many kinds of different local controllers are utilized for the injector linac component control as listed in Table 1. The ladder PLCs control the magnet power supplies, vacuum pumps,

[#] Corresponding author. *e*-mail address: masanori.satoh@kek.jp.

Table 1: List of local controllers used for the linac control system.

| Devices | Accelerator components (# of components) | # of local controllers |
|---|---|---|
| VME64x | Event based timing system (MRF EVG-230, EVR-230RF) | 25 |
| PLC | Magnet (363)<br>Vacuum (333)<br>Klystron (5)<br>Charge interlock | 59<br>26<br>5<br>3 |
| Network attached power supply | Magnet (105) | 105 |
| Linux based PLC | Profile monitor (100) | 30 |
| Embedded Linux | Klystron (66) | 66 |
| Data logger | Temperature (690) | 28 |
| Oscilloscope | BPM (90) | 23 |
| NIM modules | Timing watchdog (15) | 15 |
| Total | | 385 |

vacuum gate valves, and safety related signals. About one hundred seventy CAMAC and VME based timing delay modules have been replaced by the 25 event timing modules based on VME64x bus for the pulse-to-pulse beam modulation. The reduction of number of timing modules improves the system reliability. The PLCs used for the klystron modulators have gradually replaced by the new embedded controller of Armadillo [3]. The EPICS Input/Output controller (IOC) can run on it. The Windows based digital oscilloscope as the BPM data acquisition system will be upgraded to a new module based on VME bus for aiming at the high measurement precision less than 10 μm in the near future [4].

## EPICS ENVIRONMENT

### *Overview*

In the linac control system, the EPICS IOCs were originally implemented with the base R3.14.9 for wrapping the existing in-house control system. Table 2 shows the number of IOCs for each subsystem. During this stage, the EPICS process variables (PVs) are just communicating with the existing system based on RPC when the software access any PVs. Toward higher reliability, the server applications were remodeled for the direct communication between the IOCs and local controllers by EPICS base R3.14.12.

Table 2: Number of EPICS IOCs used for each subsystem.

| Subsystem | # of ICOs |
|---|---|
| Safety | 2 |
| Monitor | 48 |
| RF | 57 |
| Magnet | 19 |
| Vacuum | 1 |
| Operation | 3 |
| Timing | 21 |
| Temperature | 2 |
| Total | 153 |

### *Alarm system*

The independent in-house alarm applications were originally developed and utilized for each subsystems. For the comprehensive alarm management, we adopted the EPICS Control System Studio (CSS) [5] alarm 3.0.0 together with PostgreSQL 9.1.4 as a backend database engine. Currently, around 1000 PVs are registered for the CSS alarm. Figure 1 shows the Python based graphical user interface (GUI) indicating the current alarm status. In this figure, the upper half shows the summary of current alarm status. When the subgroup contains any alarmed PVs, the background color of corresponding subgroup name turns red. The bottom half shows the list of alarmed PVs in the same figure. The more detailed historical list of alarmed PVs can be also displayed in the other windows. Since the total number of registered PVs will be increased toward SuperKEKB operation, we will evaluate the speed performance and the robustness of CSS alarm with the large number of PVs.

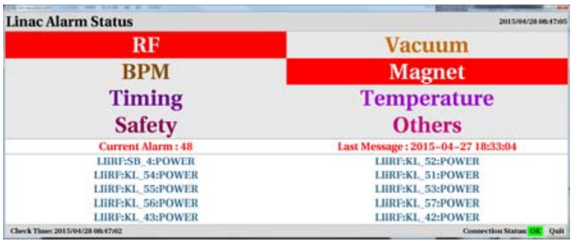


Figure 1: Example of EPICS CSS alarm status GUI.

### *Data archiver system*

For the linac original control system, we developed the simple data logging tools recording the parameters into a text file. Some different kinds of logging data viewers were developed for each parameter group. After the implementation of EPICS based control system, we employed the EPICS channel archiver and CSS archiver 3.2.2 together with PostgreSQL 9.1.4/9.3.3 as the data logging system. The total number of PVs handled for each archiver is 44063 as shown in Table 3. The daily disk space consumptions are around 2 GB and 4.5 GB for channel and CSS archiver, respectively.

The web based data browser is utilized for the CSS archiver though the standard java based viewer is used for the channel archiver. The web based one was developed by using Flex 4.6, PHP 5.3.6, and Amfphp 1.9. It can provide the easy access to archiver data from the any client devices including mobile one. It has the functions of correlation plot, multiple vertical axes, PV name search, and autocomplete for the effective parameter analysis.

The disk space consumption of CSS archiver is larger than that of channel archiver since the PostgreSQL is used as a backend database. In addition, the data retrieve speed of CSS archiver is slower than that of channel archiver. For improving these issues, the pg_reorg option was added into PostgreSQL database. This function can reduce the CSS archiver database size by 33%, and improve the data retrieving time to a certain extent. However, to achieve further improvement of data retrieving speed, we are trying to evaluate another scheme using the NoSQL type database of Casandra as a backend in the near future. In our environment of CSS archiver operation, there are still two other issues. One is that the data archiving is sometimes stopped without any error statuses. Another phenomenon is that the CSS archiver sometimes includes null data. Because of them, both of channel and CSS archivers are operated for redundant purpose.

Table 3: List of EPICS PVs registered for the EPICS channel and CSS archiver systems.

| Subgroup | # of EPICS PVs |
|---|---|
| Klystron | 1838 |
| Vacuum | 405 |
| Temperature | 694 |
| Environment | 324 |
| Magnet | 5410 |
| PF ring | 70 |
| RF phase monitor | 808 |
| Safety | 690 |
| Operation | 45 |
| Timing | 6329 |
| Alignment | 151 |
| Slow e+ facility | 30 |
| BPM | 27180 |
| Injector | 84 |

## HIGH LEVEL APPLICATIONS

### *Operator interfaces*

Many parts of original linac operator interfaces were developed by the Tcl/Tk scripting language and command line interfaces. For the rapid and intelligent application development, the Python scripting language is used for almost high level applications (HLAs). The communication between the HLAs and EPICS PVs can be established via PythonCA module developed for the KEKB project.

### *Electronic operation logbook system*

The electronic operation logbook system (OPELOG) is an important tool for the modern accelerator operations. It can enable the fast and effective shearing of machine status among accelerator operation team. In the KEK linac, the first OPELOG has been developed by using the Microsoft SQL 6.5 database in 1995. Its client GUI part has been designed by Microsoft Access and visual basic language. After the simultaneous top-up operation started, the database size drastically increased due to the frequent beam gate status change. Because of the hard maintainability and the possible registration number of data limited to 65535, we developed a new OPELOG system based on Flex 4.6, PHP 5.3.6, and Amfphp 1.9 in 2010 together with PostgreSQL 9.4.0.

Figure 2 shows an image example of the new OPELOG. It comprises the summary and detailed information screens for each operation shift. The outline of each trouble is listed by the red colored sentences in the summary screen as shown in Fig. 2 (a). The more detailed information described by an operator is displayed in another scree as shown in Fig. 2 (b). The new OPELOG can automatically record the almost routine operation state transition like the status change of beam repetition and access information into a linac tunnel. In addition, a snapshot of operation GUI image can be imported into the detailed information screen. In addition, the OPELOG has the quick search function of past log by the multiple key words and a specific time period. It can be a strong help to find a solution of similar trouble occurred in the past.

(a) Summary screen. (b) Detailed screen.

Figure 2: Image example of the linac operation electronic logbook system based on web application.

## CONCLUSION

Since the middle phase of KEKB operation, the linac beam control system has been gradually transferred from the in-house system based on RPC to the EPICS based one. The speed performance improvement of archiver and the development of advanced commissioning tools are big issues toward SuperKEKB operation. In addition, we will develop the virtual computing server based on the VMware vCenter for aiming at the much higher reliability.

## REFERENCES


[1] http://www.aps.anl.gov/epics/.

[2] K. Furukawa et al., “Pulse-to-pulse Beam Modulation and Event-based Beam Feedback Systems at KEKB Linac”, in Proceedings of IPAC’10, Kyoto, Japan, pp.1271-1273 (2010).

[2] http://ics-web.sns.ornl.gov/kasemir/archiver/.

[3] Y. Yano et al., “RF CONTROL SYSTEM FOR SUPERKEKB INJECTOR LINAC”, in Proceedings of the 11th Annual Meeting of Particle Accelerator Society of Japan, August 9-11 2014, Aomori, Japan, pp.624-628 (2014).

[4] R. Ichimiya et al., “DEVELOPMENT OF HIGH PRECISION BEAM POSITION MONITOR READOUT SYSTEM WITH NARROW BANDPASS FILTERS FOR THE KEKB INJECTOR LINAC TOWARDS THE SUPERKEKB”, in Proceedings of IBIC2013, Shanghai, China, September 16-19, pp.698-701 (2013).

[5] http://controlsystemstudio.org/.